Gröbner Basis and deducibility

Charlie Jacomme
July 13, 2018
LSV & LORIA\textsuperscript{a}

\textsuperscript{a}CNRS, INRIA, ENS Paris-Saclay
Introduction to security
A rising need for security

Confidentiality:
- Banking operations
- Smartphones with GPS
- ...

Authentication:
- Internet shopping
- Private emails
- ...

A rising need for security

Confidentiality:
- Banking operations
- Smartphones with GPS
- ...

Authentication:
- Internet shopping
- Private emails
- ...

A public encryption scheme:

- A public key $pk$
- A secret key $sk$
- An encryption function $enc(\text{message}, pk)$
- A decryption function $dec(\text{cypher}, sk)$

based on exponentiation in group and hardness of discrete logarithm
A public encryption scheme:

- A public key $pk$
- A secret key $sk$
A public encryption scheme:

- A public key $pk$
- A secret key $sk$
- An encryption function $enc(message, pk)$
- A decryption function $dec(cypher, sk)$
A public encryption scheme:

- A public key \( pk \)
- A secret key \( sk \)
- An encryption function \( enc(message, pk) \)
- A decryption function \( dec(cypher, sk) \)

based on exponentiation in group and hardness of discrete logarithm
Protocols

Alice

$sk_A, secret$

Bob

$sk_B$

$\langle secret \rangle_{pk_B}, Alice$

$\langle secret \rangle_{pk_A}, Bob$
Protocols

Alice

$sk_A, secret$

$\langle secret \rangle_{pk_B}, Alice$

Bob

$sk_B$

$\langle secret \rangle_{pk_A}, Bob$

Is secret secret?
Protocols

Alice

\(sk_A, secret\)

\(\langle secret\rangle_{pk_B}, Alice\)

\(\langle secret\rangle_{pk_B}, Bob\)

Charlie

\(sk_C\)

\(\langle secret\rangle_{pk_B}, Charlie\)

Bob

\(sk_B\)

\(\langle secret\rangle_{pk_C}, Bob\)

No!
Formal methods

Proofs of security are

- difficult
- error prone
Formal methods

Proofs of security are

- difficult
- error prone

→ We want automation
Deducibility
Deducibility

Given a set of messages, can an attacker deduce a secret?
Deducibility

Given a set of messages, can an attacker deduce a secret?

Messages are

- $x, y, z, \ldots$ random variables over $\mathbb{K}$
- $g^f$ with $f \in \mathbb{K}[X]$
Deducibility

Given a set of messages, can an attacker deduce a secret?

Messages are

- $x, y, z, \ldots$ random variables over $\mathbb{K}$
- $g^f$ with $f \in \mathbb{K}[X]$

Example

$x, g^{xy}, g^{y^2} \vdash g^{y^2+y}$
Deducibility

Given a set of messages, can an attacker deduce a secret?

Messages are

• $x, y, z, \ldots$ random variables over $\mathbb{K}$
• $g^f$ with $f \in \mathbb{K}[X]$

Example

\[
x, g^{xy}, g^y^2 \vdash g^{y^2+y}
\]

\[
g^{y^2+y} = (g^{xy})^{x^{-1}} \times g^y^2
\]
Our generalized problem

\[ \Gamma \models X, g^{f_1}, \ldots, g^{f_k} \vdash g^h \]

\{ \Gamma \text{ axioms} \\ X \text{ public variables} \\ Y \text{ secret variables} \\ g \text{ group} \\ f_i, h \text{ polynomials over } \mathbb{K}[X, Y] \}
The algorithm

\[ \Gamma \models X, g_1^{f_1}, \ldots, g_k^{f_k} \vdash g_t^h \]

With Gröbner Basis
The algorithm

\[ \Gamma \models X, g^{f_1}, \ldots, g^{f_k} \vdash g_t^h \]

With Gröbner Basis

1. Characterize the attacker knowledge:

\[ M = \left\{ \sum_i e_i \times f_i \mid e_i \in \mathbb{K}[X] \right\} \]
The algorithm

\[ \Gamma \models X, g^{f_1}, \ldots, g^{f_k} \vdash g^h \]

With Gröbner Basis

1. Characterize the attacker knowledge:

\[ M = \left\{ \sum_i e_i \times f_i \mid e_i \in \mathbb{K}[X] \right\} \]

2. Saturate using the axioms, if \( \Gamma = \{ p_k \neq 0 \} \):

\[ M :_{\mathbb{K}[X,Y]} (p_1\ldots p_n)^\infty = \left\{ f \in \mathbb{K}[X,Y] \mid \exists n \in \mathbb{N}, f \times (p_1\ldots p_n)^n \in M \right\} \]
The algorithm

\[ \Gamma \models X, g^{f_1}, \ldots, g^{f_k} \vdash g^h \]

With Gröbner Basis

1. Characterize the attacker knowledge:

\[ M = \left\{ \sum_{i} e_i \times f_i \middle| e_i \in \mathbb{K}[X] \right\} \]

2. Saturate using the axioms, if \( \Gamma = \{ p_k \neq 0 \} \):

\[ M :_{\mathbb{K}[X, Y]} (p_1 \ldots p_n)\infty = \left\{ f \in \mathbb{K}[X, Y] \middle| \exists n \in \mathbb{N}, f \times (p_1 \ldots p_n)^n \in M \right\} \]

3. Test the membership.
Gröbner Basis
For $f, g_i$ in $\mathbb{K}[X, Y]$, we have:

- an ordering on monomials
- if $lm(f) = q lm(g_1)$ then $\text{red}_1^X(f, g_1) = f - qg$
- $\text{red}^X(f, g_i)$ is the iteration of $\text{red}_1$ for all $g_i$
(in)Formal definition

For $f, g_i$ in $\mathbb{K}[X, Y]$, we have:

- an ordering on monomials
- if $\text{lm}(f) = q\text{lm}(g_1)$ then $\text{red}_1^X(f, g_1) = f - qg$
- $\text{red}_1^X(f, g_i)$ is the iteration of $\text{red}_1$ for all $g_i$

Gröbner Basis

$G = \{g_i\}$ is a GB pf $M$ iff $\forall h \in M, \text{red}(h, G) = 0$
For $f, g_i$ in $\mathbb{K}[X, Y]$, we have:

$$S(f, g_1) = \frac{lppcm(f, g_1)}{lm(g)} f - \frac{lppcm(f, g_1)}{lm(f)} g$$
Buchberger’s algorithm

For $f, g_i$ in $\mathbb{K}[X, Y]$, we have:

$$S(f, g_1) = \frac{lppcm(f, g_1)}{lm(g)} f - \frac{lppcm(f, g_1)}{lm(f)} g$$

Given $G = \{g_i\}$:

- compute a $S(g_i, g_j)$
- reduce it w.r.t to $G$
- if its remainder is non zero, add it to $G$
Buchberger’s algorithm

For \( f, g_i \) in \( \mathbb{K}[X, Y] \), we have:

\[
S(f, g_1) = \frac{lppcm(f, g_1)}{lm(g)} f - \frac{lppcm(f, g_1)}{lm(f)} g
\]

Given \( G = \{g_i\} \):

- compute a \( S(g_i, g_j) \)
- reduce it w.r.t to \( G \)
- if its remainder is non zero, add it to \( G \)
Goal

If $M = \langle G \rangle$, compute the GB of

$$M : \mathbb{K}[X, Y] (p)^\infty = \{ f \in \mathbb{K}[X, Y] | \exists n \in \mathbb{N}, f \times (p)^n \in M \}$$
Saturation

Goal
If $M = \langle G \rangle$, compute the GB of

$$M : \mathbb{K}[X, Y] (p)_{\infty} = \{ f \in \mathbb{K}[X, Y] | \exists n \in \mathbb{N}, f \times (p)^n \in M \}$$

Magic trick

- compute the GB of $G \cup (1 - tp)$, with $t$ a fresh variable
- keep only the base element not containing $t$. 
The algorithm

\[ \Gamma \models X, g^{f_1}, \ldots, g^{f_k} \models g^h \]

With Gröbner Basis

1. Characterize the attacker knowledge:

   \[ M = \{ \sum_i e_i \times f_i \mid e_i \in \mathbb{K}[X] \} \]

2. Saturate using the axioms, if \( \Gamma = \{ p_k \neq 0 \} \):

   \[ M :_{\mathbb{K}[X,Y]} (p_1 \ldots p_n)^\infty = \{ f \in \mathbb{K}[X,Y] \mid \exists n \in \mathbb{N}, f \times (p_1 \ldots p_n)^n \in M \} \]

3. Test the membership.
Conclusion
It is a good idea to have general knowledge in math when doing computer science!